FIG. 1

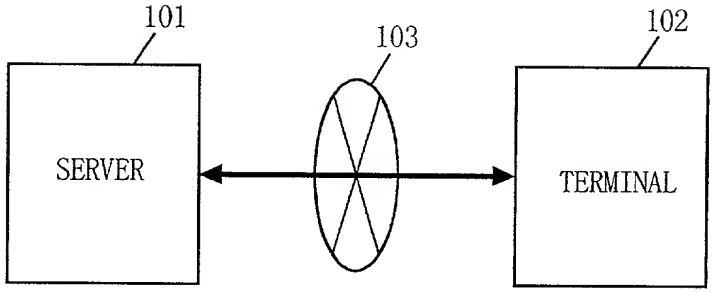

101
103
102
SERVER
TERMINAL

FIG. 2

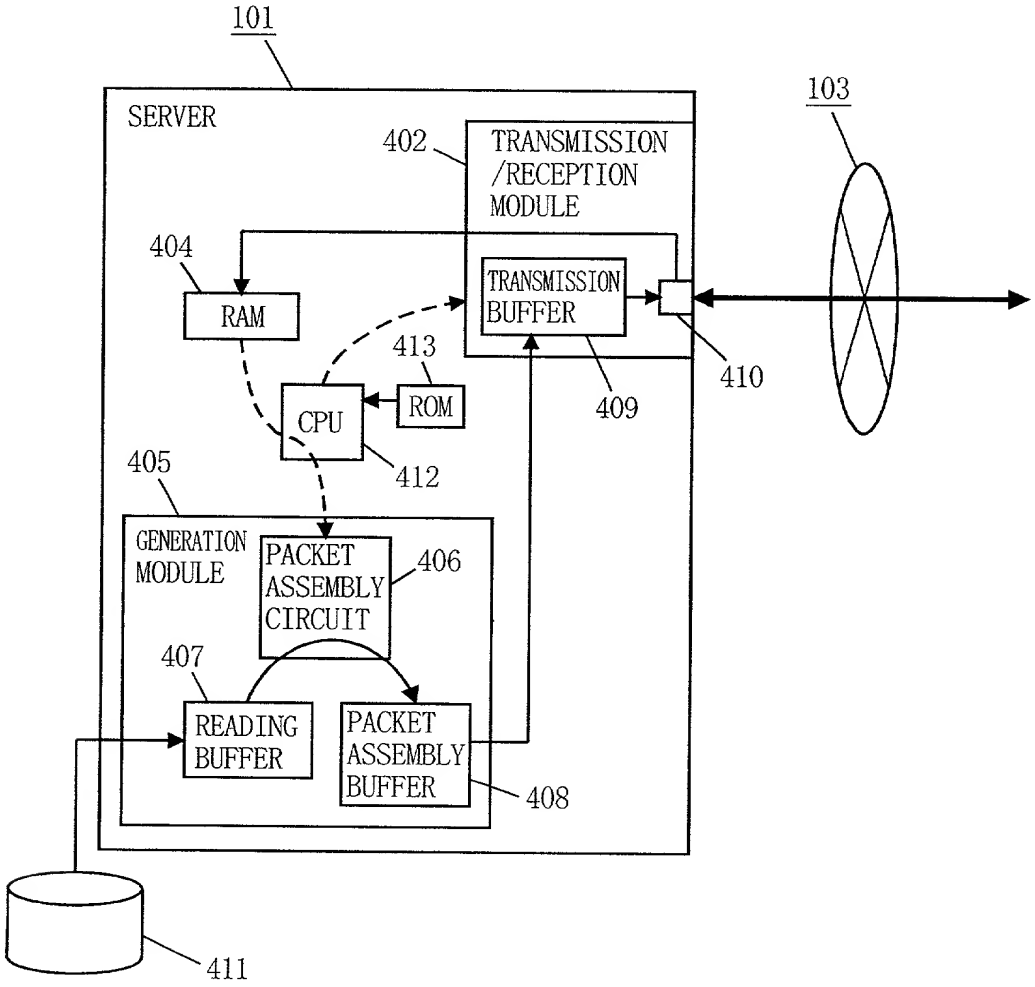

101
SERVER
402
TRANSMISSION /RECEPTION MODULE
103
404
RAM
TRANSMISSION BUFFER
410
413
ROM
CPU
409
412
405
GENERATION MODULE
PACKET ASSEMBLY CIRCUIT
406
407
READING BUFFER
PACKET ASSEMBLY BUFFER
408
411

FIG. 3

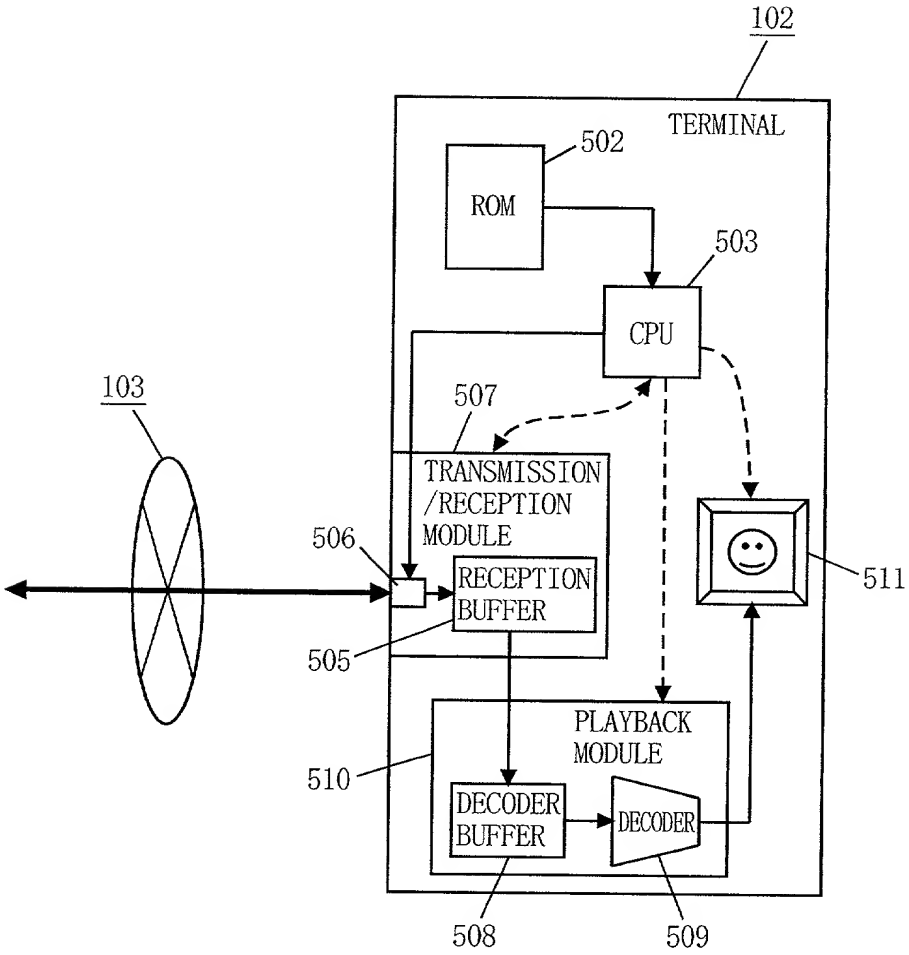

102
TERMINAL
502
ROM
503
CPU
103
507
TRANSMISSION
/RECEPTION
MODULE
506
RECEPTION
BUFFER
505
511
PLAYBACK
MODULE
510
DECODER
BUFFER
DECODER
508
509

FIG. 4

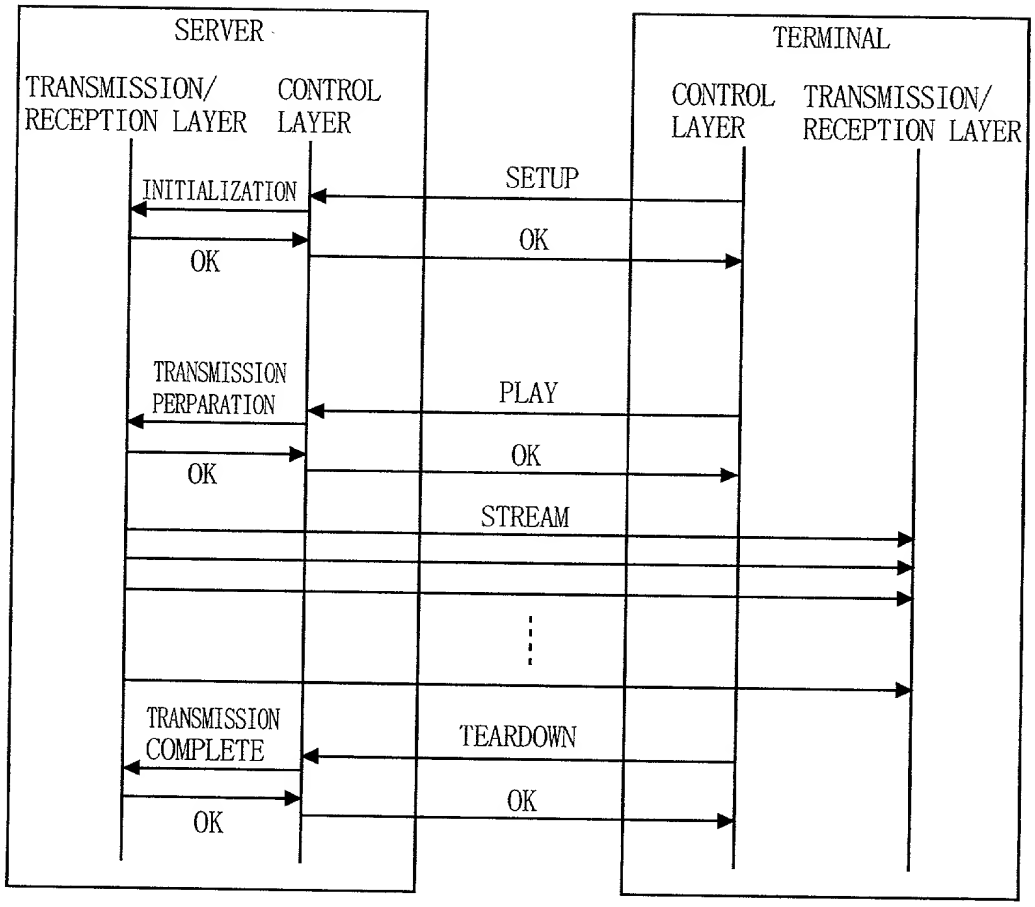

SERVER
TRANSMISSION/ RECEPTION LAYER
CONTROL LAYER
TERMINAL
CONTROL LAYER
TRANSMISSION/ RECEPTION LAYER
SETUP
INITIALIZATION
OK
OK
TRANSMISSION PERPARATION
PLAY
OK
OK
STREAM
TRANSMISSION COMPLETE
TEARDOWN
OK
OK

FIG. 5

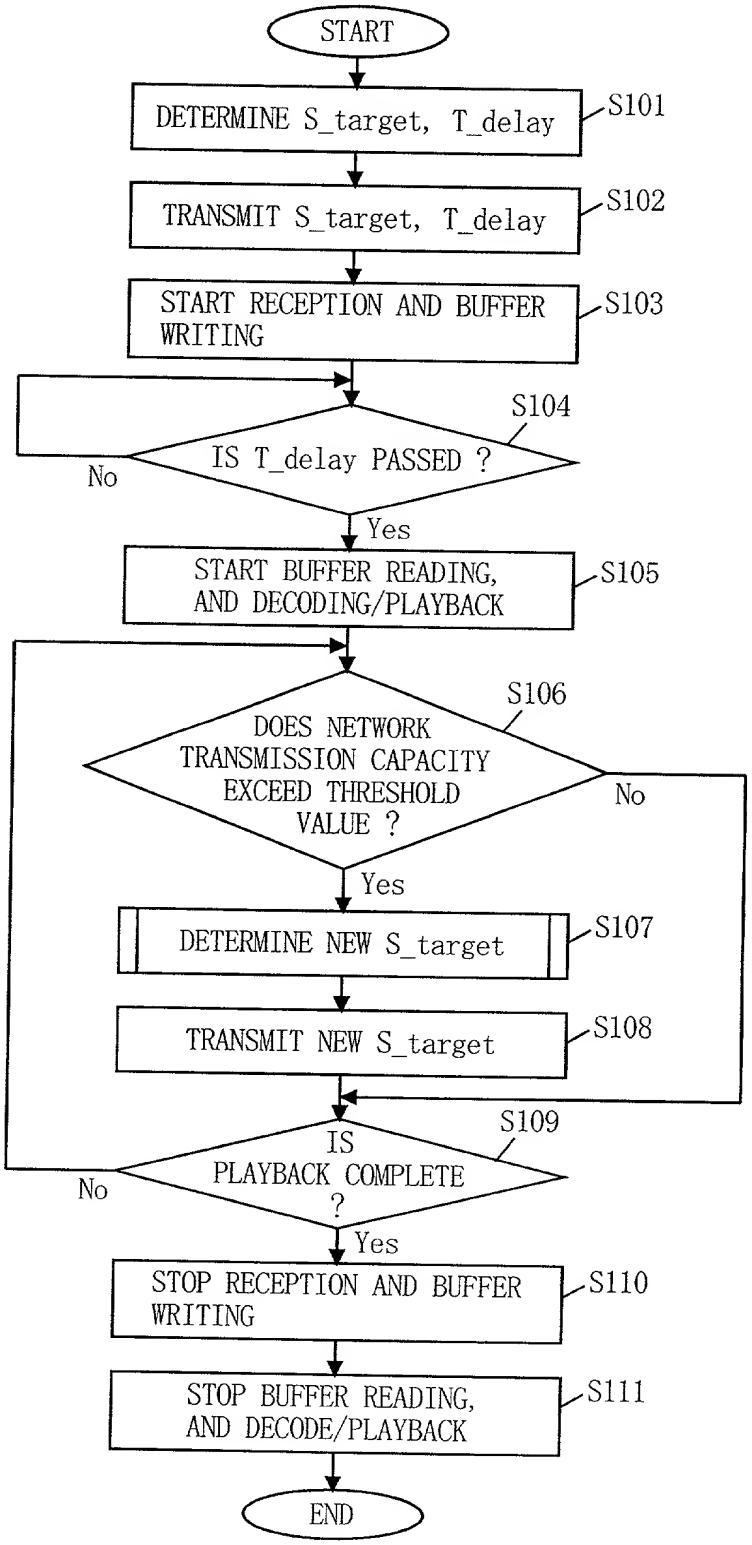

START
DETERMINE S_target, T_delay
S101
TRANSMIT S_target, T_delay
S102
START RECEPTION AND BUFFER WRITING
S103
S104
IS T_delay PASSED ?
No
Yes
START BUFFER READING, AND DECODING/PLAYBACK
S105
S106
DOES NETWORK TRANSMISSION CAPACITY EXCEED THRESHOLD VALUE ?
No
Yes
DETERMINE NEW S_target
S107
TRANSMIT NEW S_target
S108
S109
IS PLAYBACK COMPLETE ?
No
Yes
STOP RECEPTION AND BUFFER WRITING
S110
STOP BUFFER READING, AND DECODE/PLAYBACK
S111
END

FIG. 6

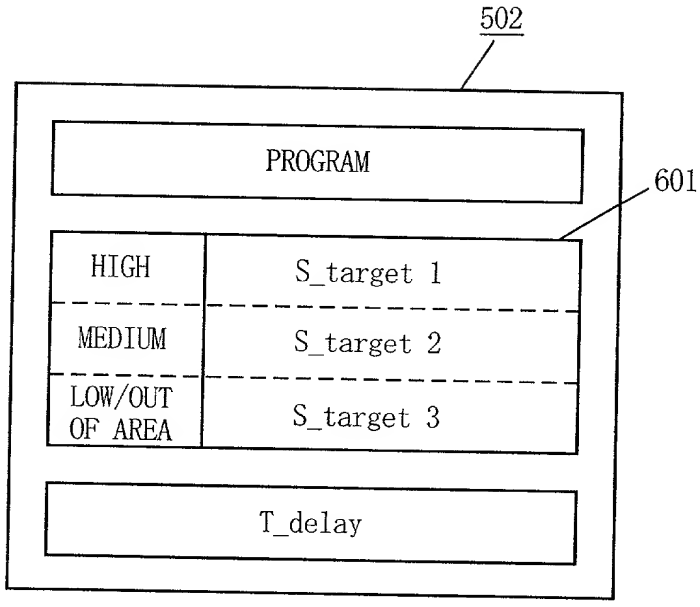

502
PROGRAM
601
HIGH
S_target 1
MEDIUM
S_target 2
LOW/OUT OF AREA
S_target 3
T_delay

FIG. 7A

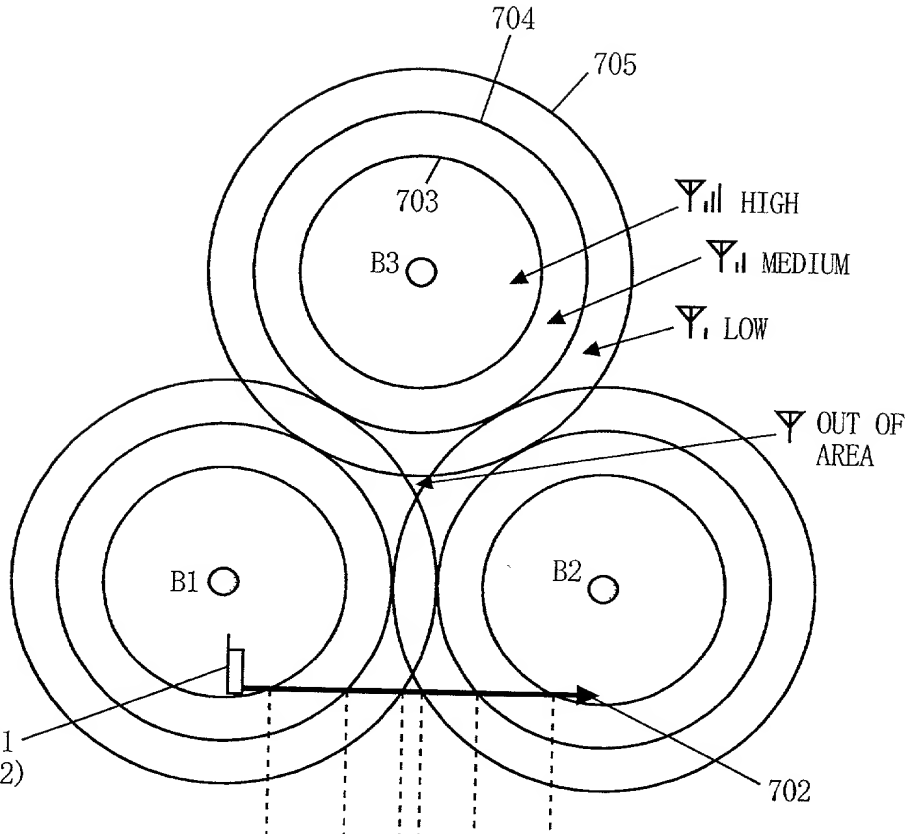

704
705
703
B3
HIGH
MEDIUM
LOW
OUT OF AREA
B1
B2
701
(102)
702

FIG. 7B

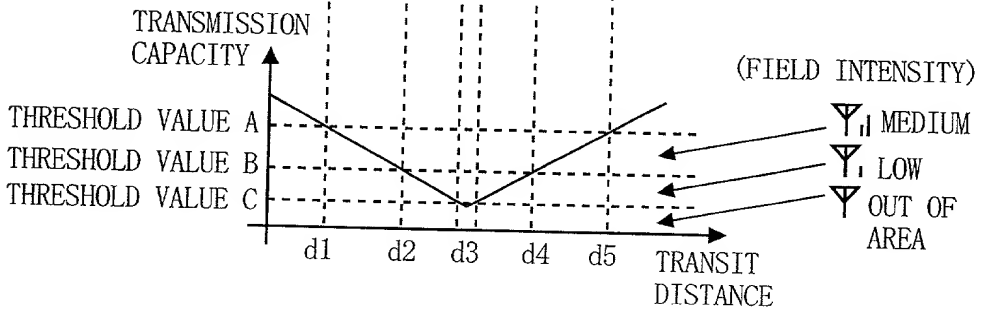

TRANSMISSION CAPACITY
(FIELD INTENSITY)
THRESHOLD VALUE A
THRESHOLD VALUE B
THRESHOLD VALUE C
MEDIUM
LOW
OUT OF AREA
d1
d2
d3
d4
d5
TRANSIT DISTANCE

FIG. 8

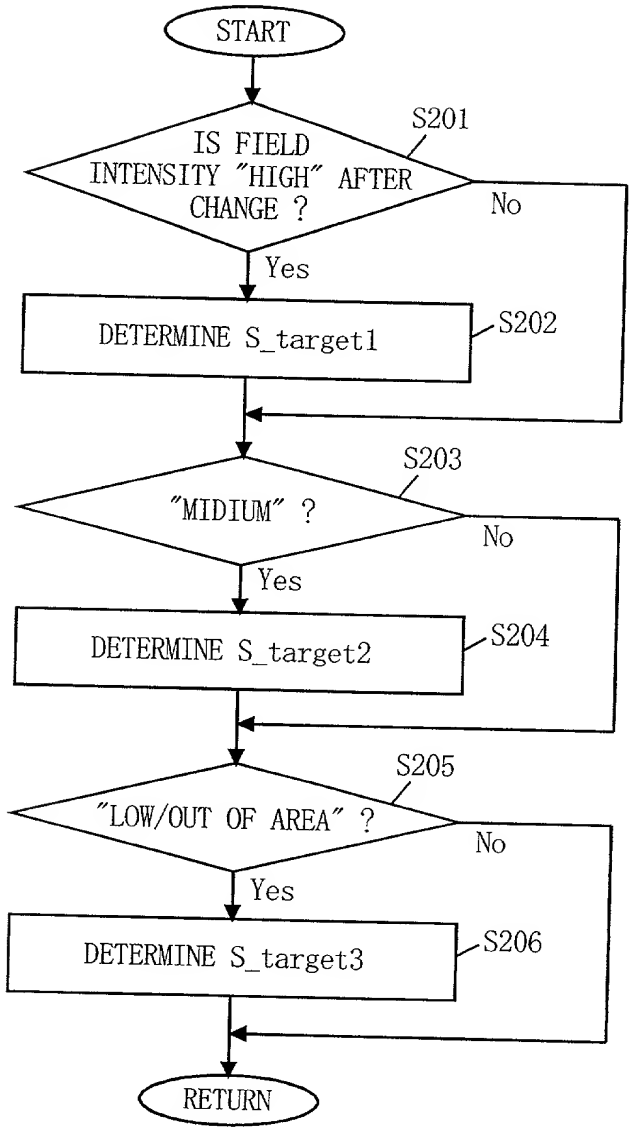

START
S201
IS FIELD INTENSITY "HIGH" AFTER CHANGE ?
No
Yes
DETERMINE S_target1
S202
S203
"MIDIUM" ?
No
Yes
DETERMINE S_target2
S204
S205
"LOW/OUT OF AREA" ?
No
Yes
DETERMINE S_target3
S206
RETURN

FIG. 9

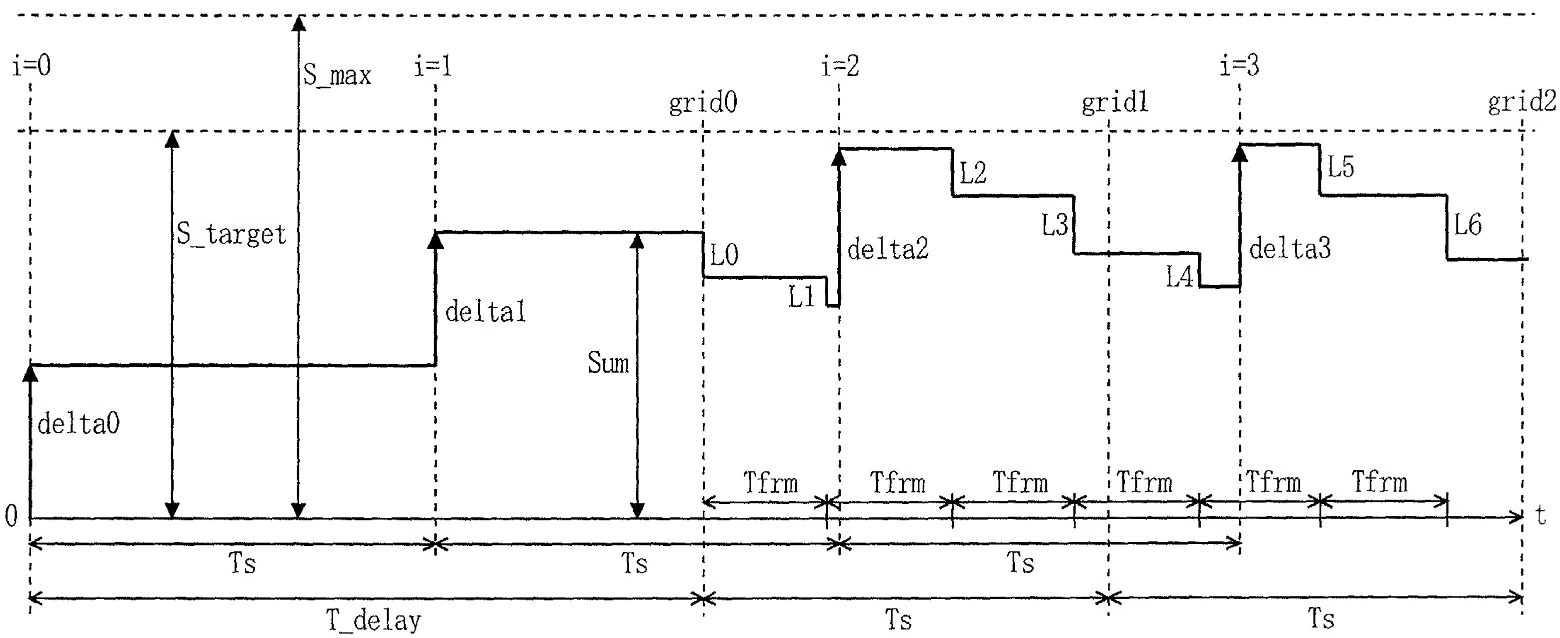


FIG. 10

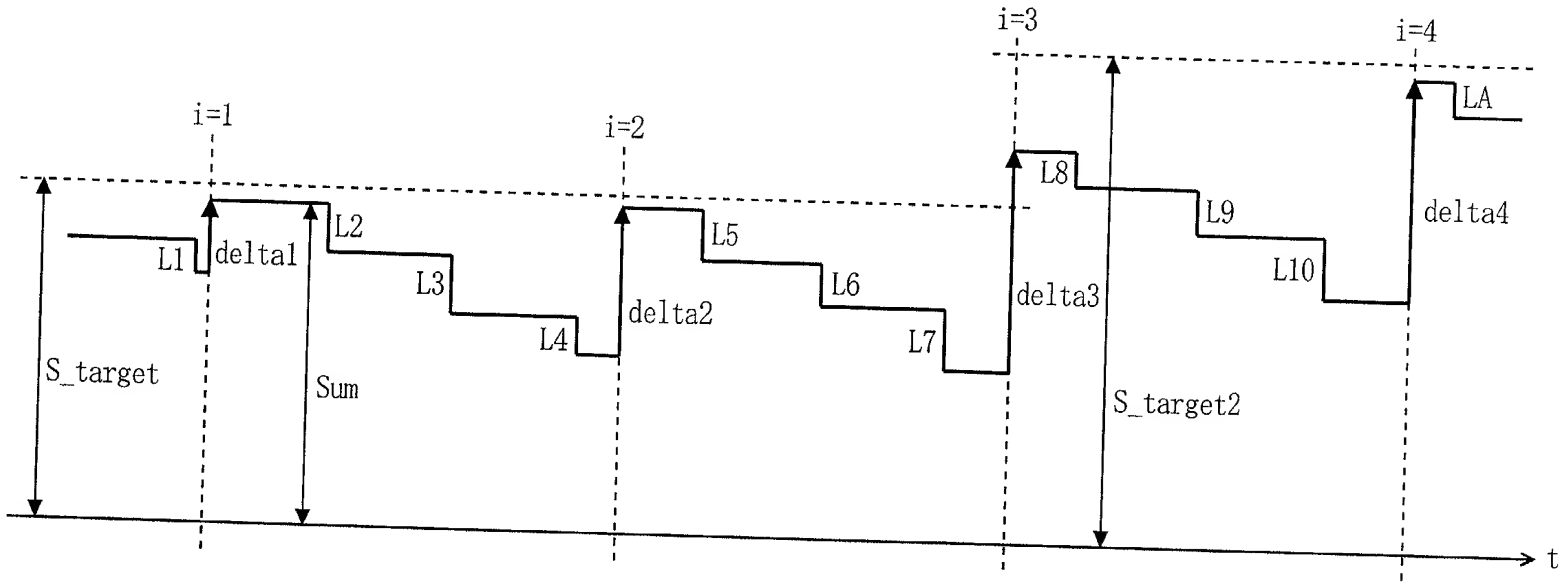

i=1
i=2
i=3
i=4
L1
delta1
L2
L3
L4
delta2
L5
L6
L7
delta3
L8
L9
L10
delta4
LA
S_target
Sum
S_target2
t

FIG. 11

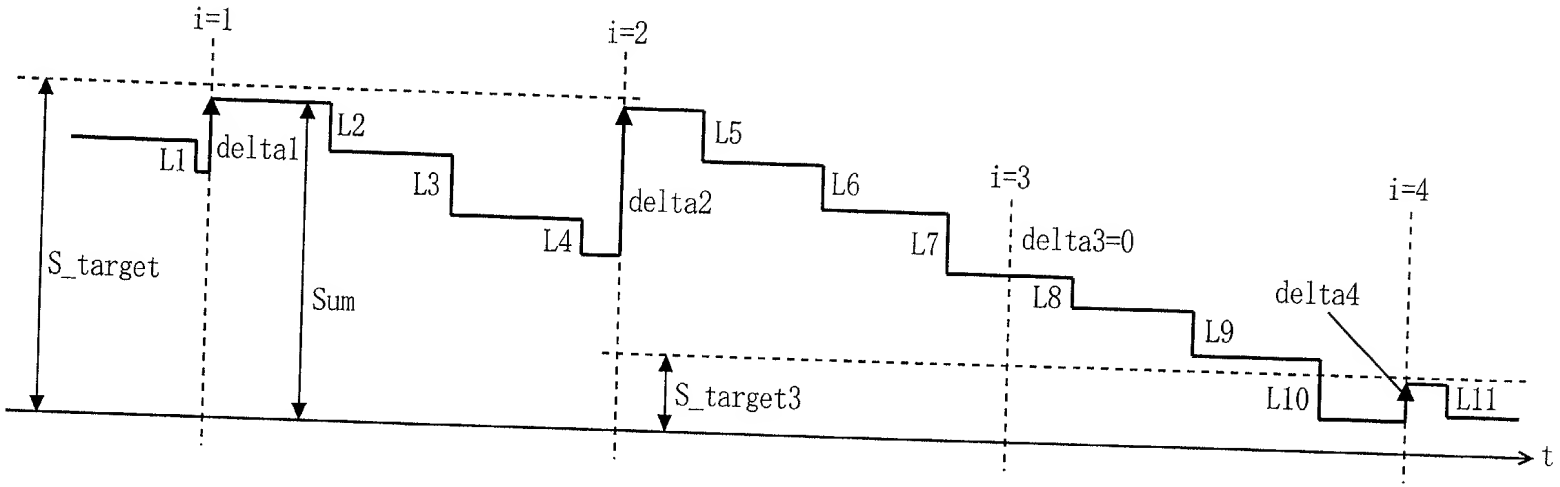

i=1
i=2
i=3
i=4
L1
deltal
L2
L3
L4
delta2
L5
L6
L7
delta3=0
L8
L9
delta4
L10
L11
S_target
Sum
S_target3
t

FIG. 12

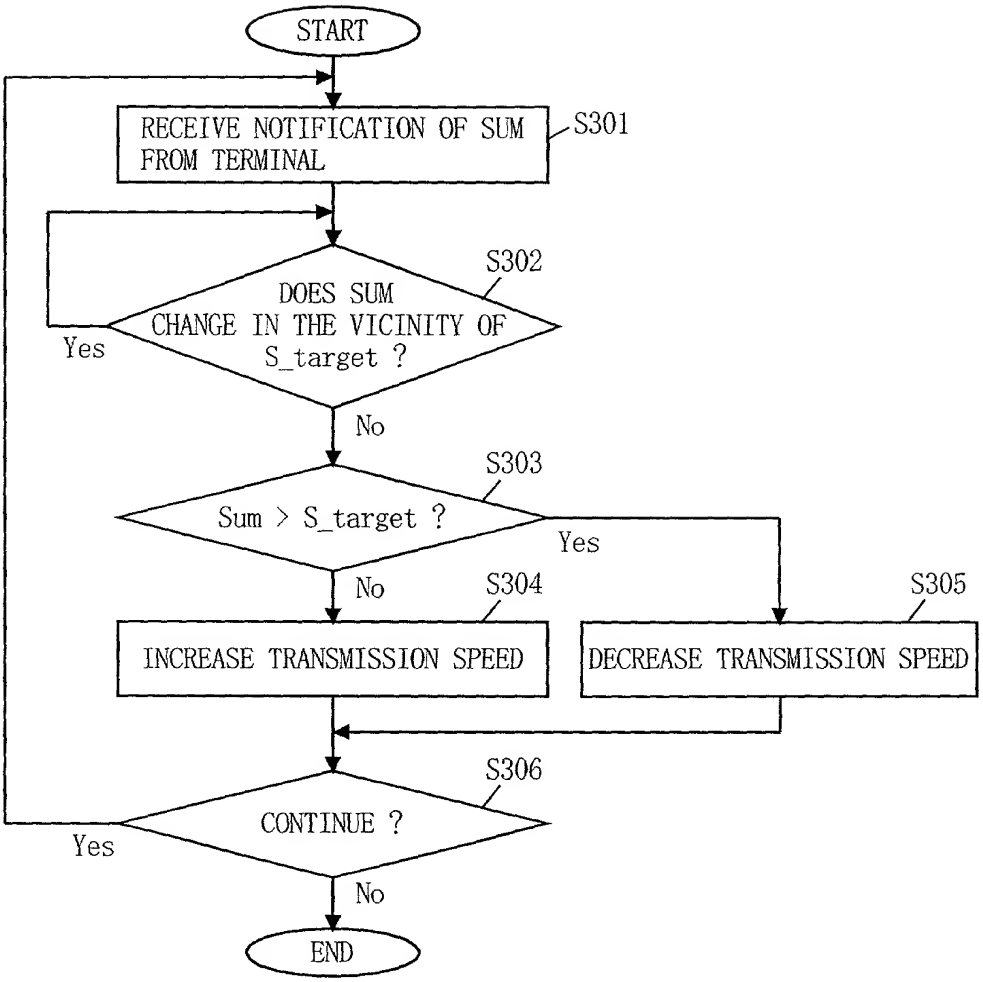

START
RECEIVE NOTIFICATION OF SUM FROM TERMINAL
S301
DOES SUM CHANGE IN THE VICINITY OF S_target ?
S302
Yes
No
Sum > S_target ?
S303
Yes
No
INCREASE TRANSMISSION SPEED
S304
DECREASE TRANSMISSION SPEED
S305
CONTINUE ?
S306
Yes
No
END

FIG. 13

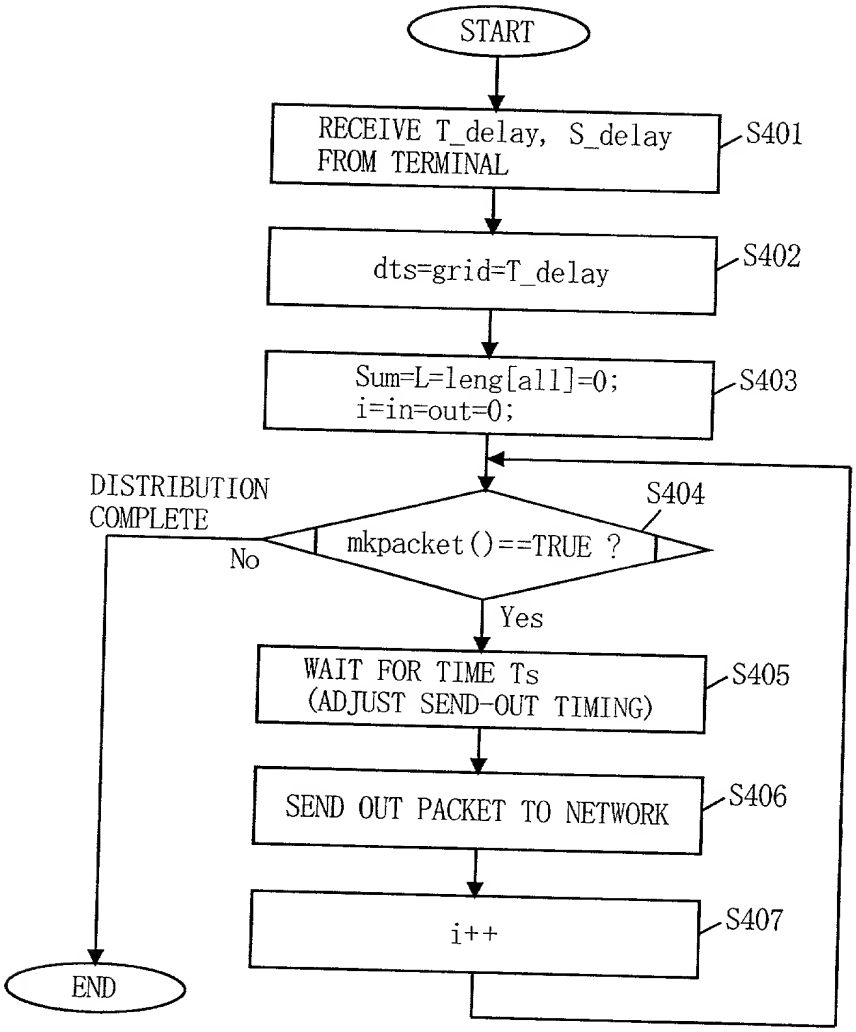
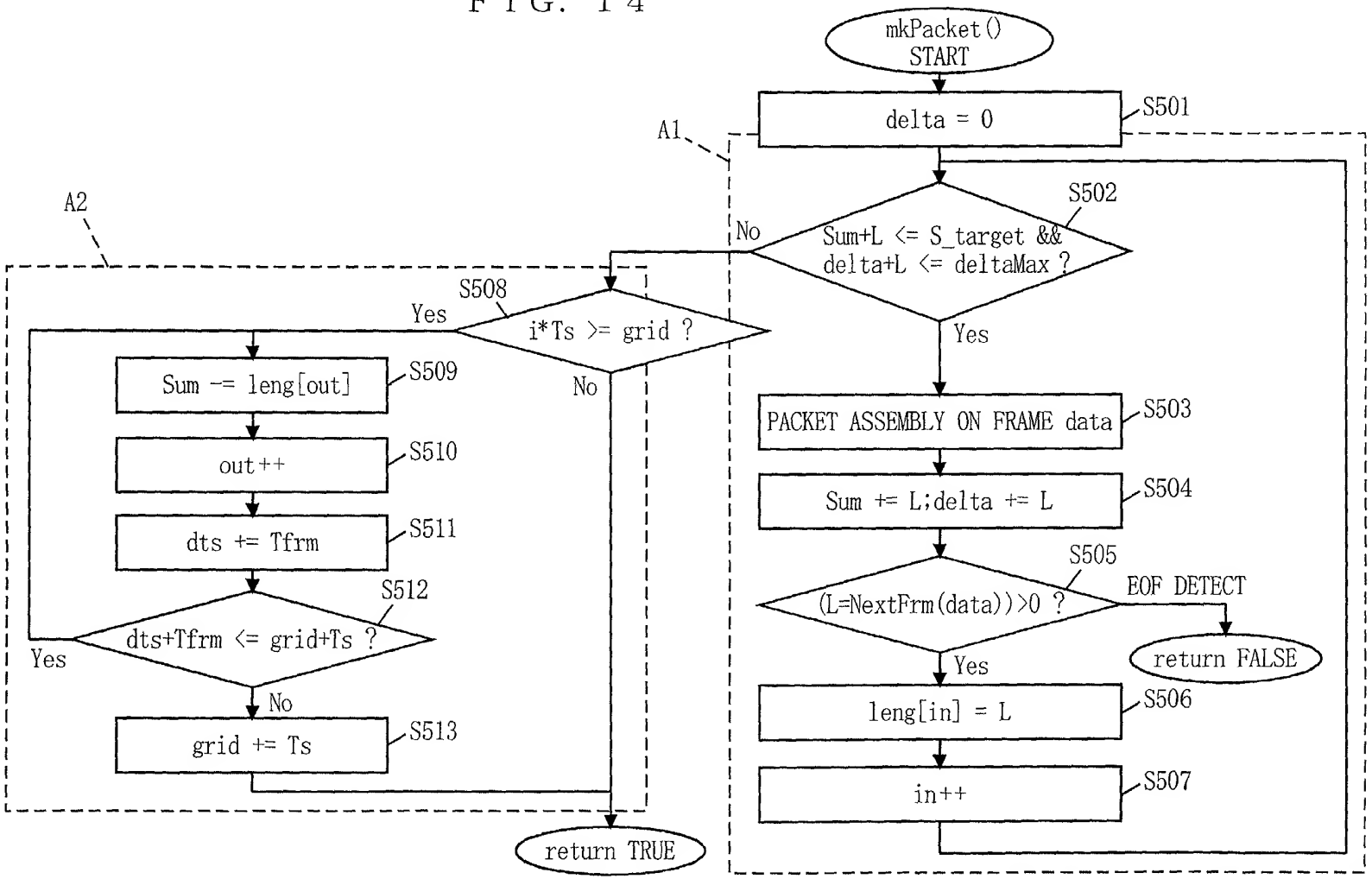
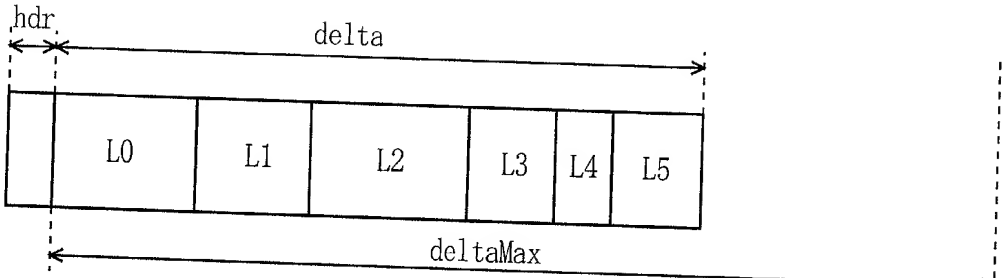

START
RECEIVE T_delay, S_delay FROM TERMINAL
S401
dts=grid=T_delay
S402
Sum=L=leng[all]=0;
i=in=out=0;
S403
DISTRIBUTION COMPLETE
S404
mkpacket()==TRUE ?
No
Yes
WAIT FOR TIME Ts (ADJUST SEND-OUT TIMING)
S405
SEND OUT PACKET TO NETWORK
S406
i++
S407
END

FIG. 14

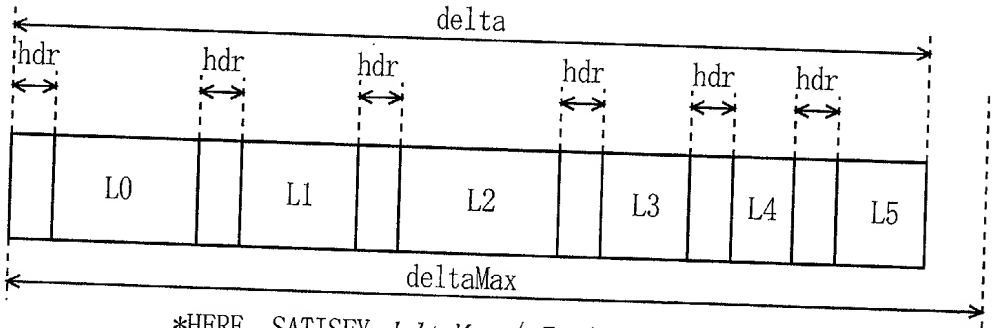

mkPacket()
START
delta = 0
S501
A1
S502
Sum+L <= S_target &&
delta+L <= deltaMax ?
No
Yes
PACKET ASSEMBLY ON FRAME data
S503
Sum += L;delta += L
S504
S505
(L=NextFrm(data))>0 ?
EOF DETECT
return FALSE
Yes
leng[in] = L
S506
in++
S507
A2
S508
i*Ts >= grid ?
Yes
No
Sum -= leng[out]
S509
out++
S510
dts += Tfrm
S511
S512
dts+Tfrm <= grid+Ts ?
Yes
No
grid += Ts
S513
return TRUE

(A) of FIG. 15



*HERE, SATISFY (*deltaMax* + *hdr*)/ *Ts* < *NetworkRate*

(B) of FIG. 15



*HERE, SATISFY *deltaMax* / *Ts* < *NetworkRate*

FIG. 16

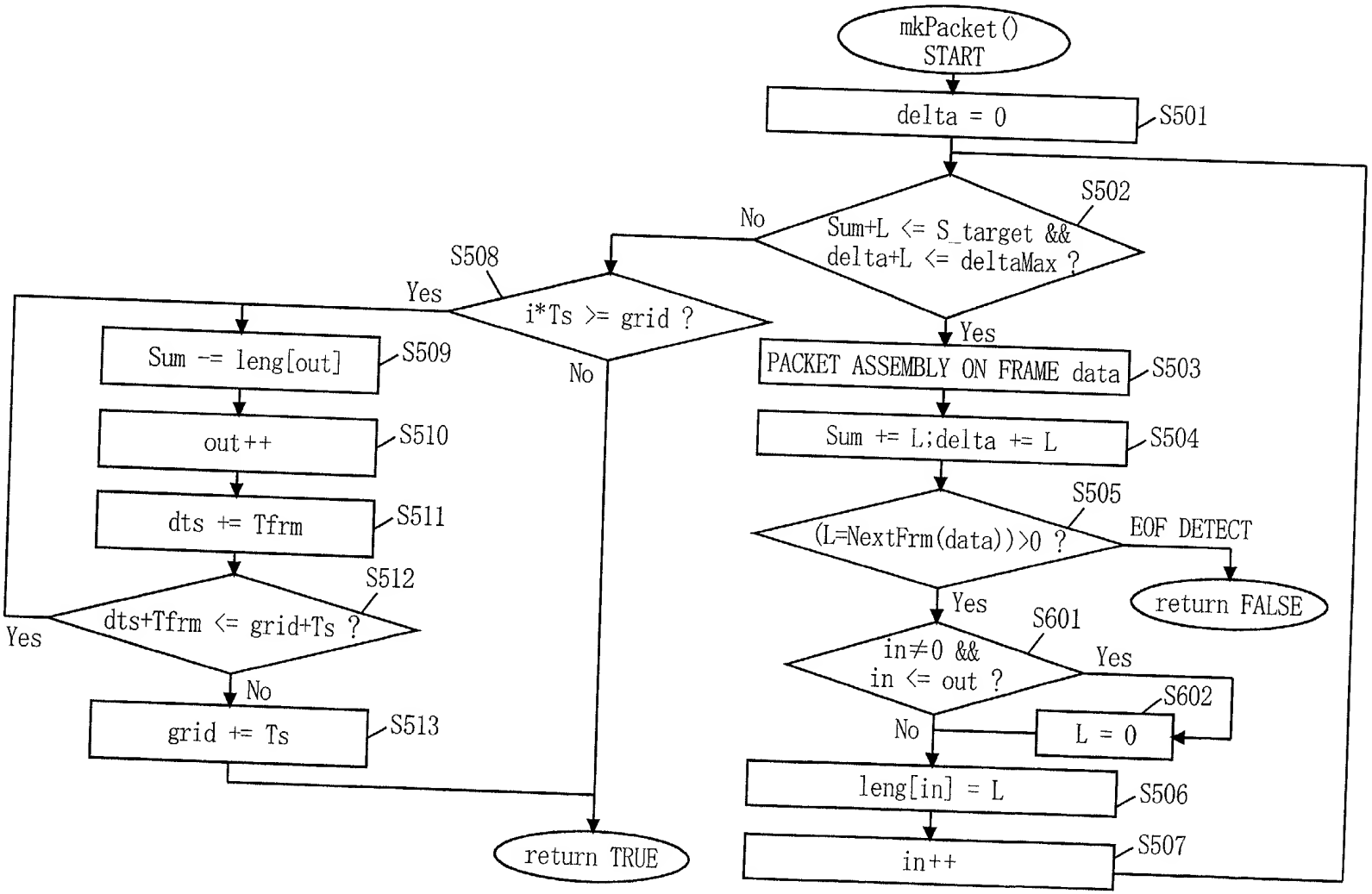

mkPacket()
START
delta = 0
S501
S502
Sum+L <= S_target &&
delta+L <= deltaMax ?
No
Yes
S503
PACKET ASSEMBLY ON FRAME data
S504
Sum += L;delta += L
S505
(L=NextFrm(data))>0 ?
EOF DETECT
return FALSE
Yes
S601
in≠0 &&
in <= out ?
Yes
S602
L = 0
No
S506
leng[in] = L
S507
in++
S508
i*Ts >= grid ?
Yes
No
S509
Sum -= leng[out]
S510
out++
S511
dts += Tfrm
S512
dts+Tfrm <= grid+Ts ?
Yes
No
S513
grid += Ts
return TRUE

FIG. 17

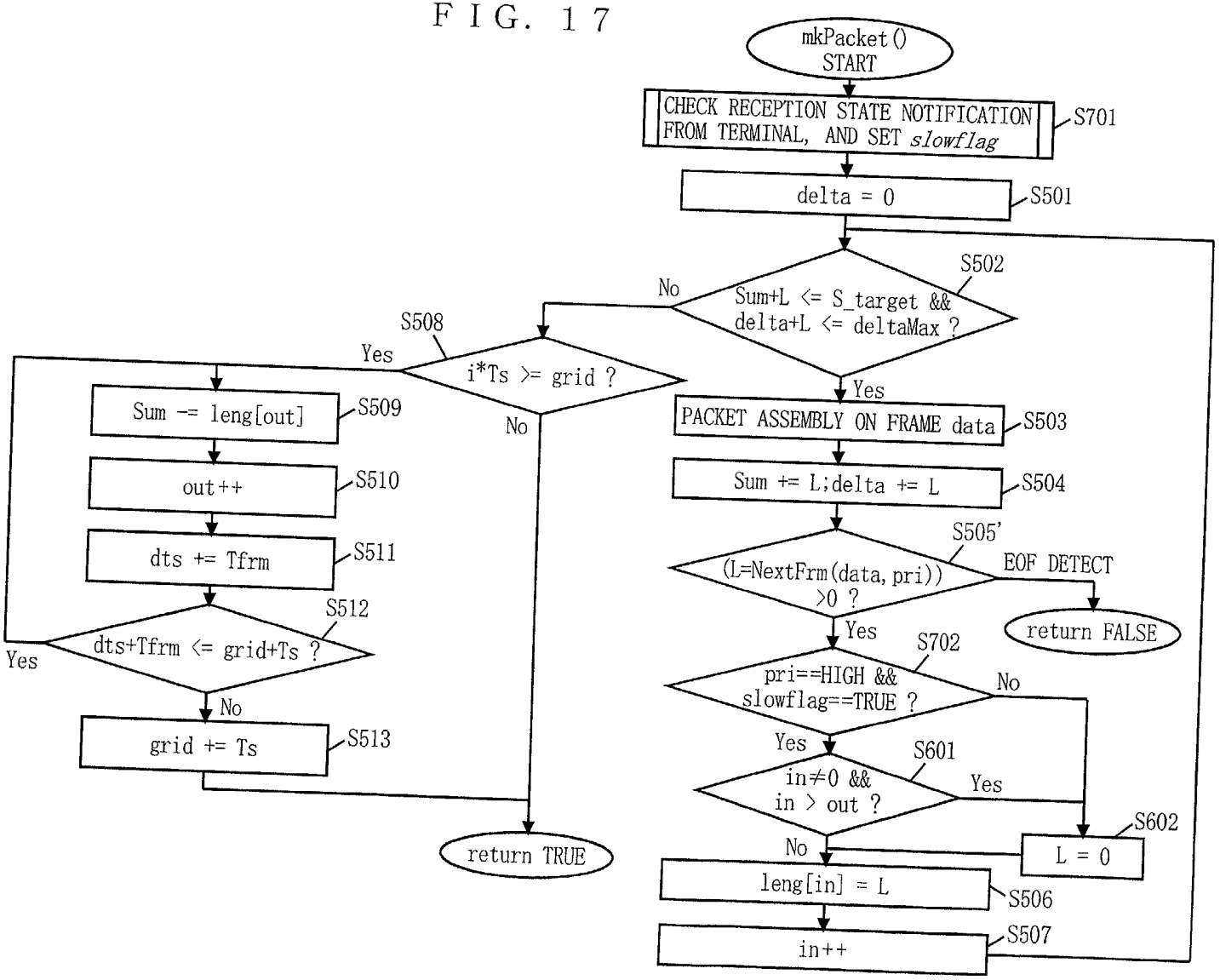

mkPacket()
START
CHECK RECEPTION STATE NOTIFICATION FROM TERMINAL, AND SET slowflag
S701
delta = 0
S501
S502
Sum+L <= S_target && delta+L <= deltaMax ?
No
Yes
S508
i*Ts >= grid ?
Yes
No
Sum -= leng[out]
S509
out++
S510
dts += Tfrm
S511
S512
dts+Tfrm <= grid+Ts ?
Yes
No
grid += Ts
S513
return TRUE
PACKET ASSEMBLY ON FRAME data
S503
Sum += L;delta += L
S504
S505'
(L=NextFrm(data, pri)) >0 ?
EOF DETECT
return FALSE
Yes
S702
pri==HIGH && slowflag==TRUE ?
No
Yes
S601
in≠0 && in > out ?
Yes
No
S602
L = 0
leng[in] = L
S506
in++
S507

FIG. 18

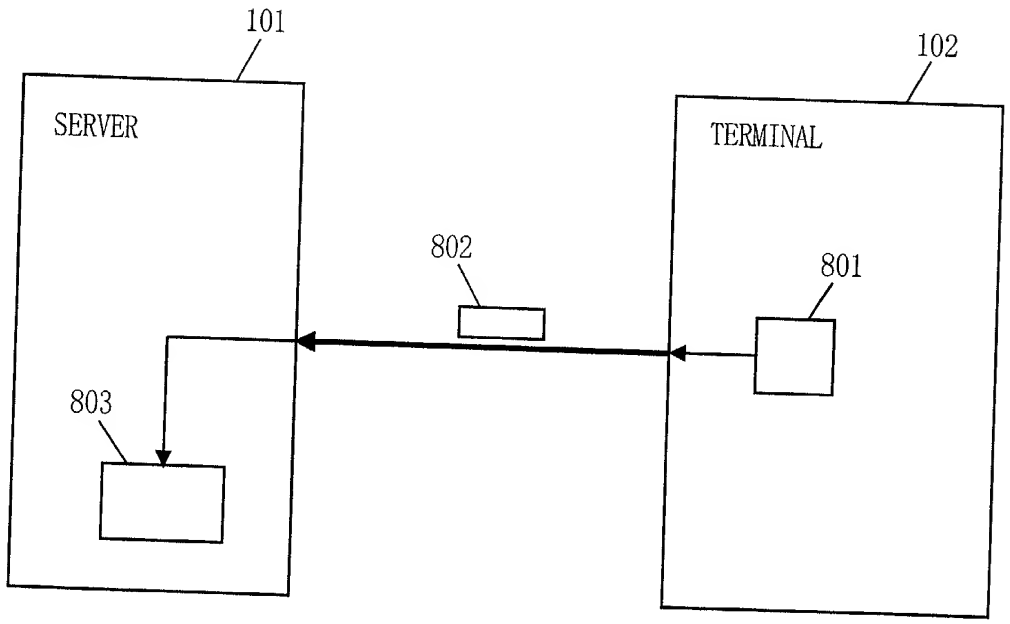


FIG. 19A
PRIOR ART

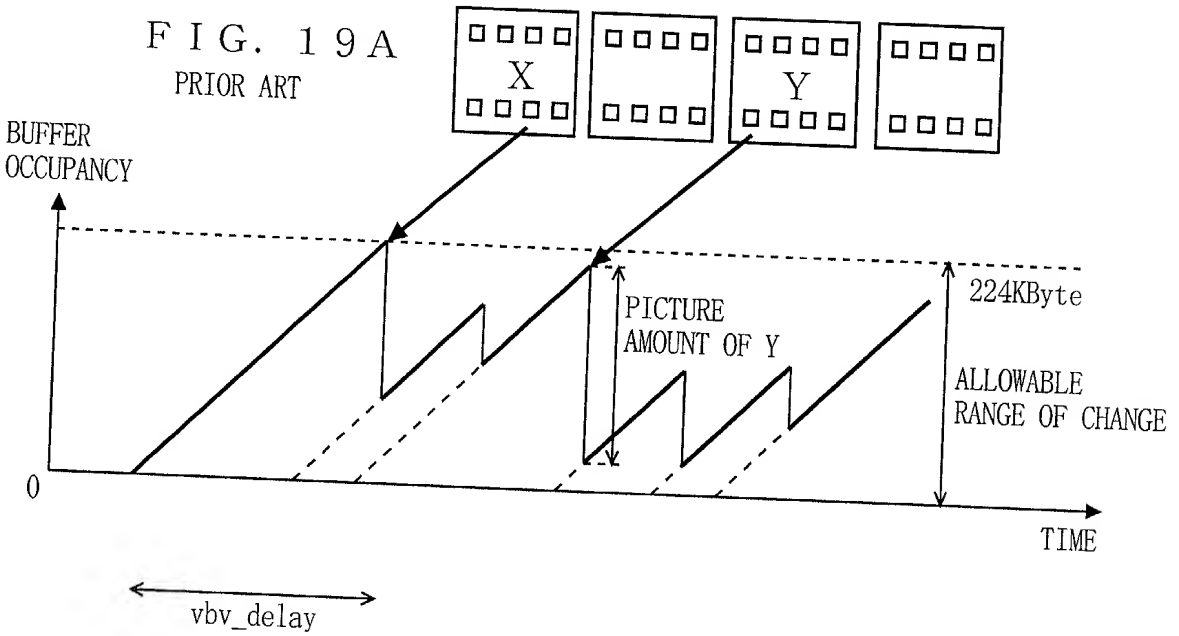
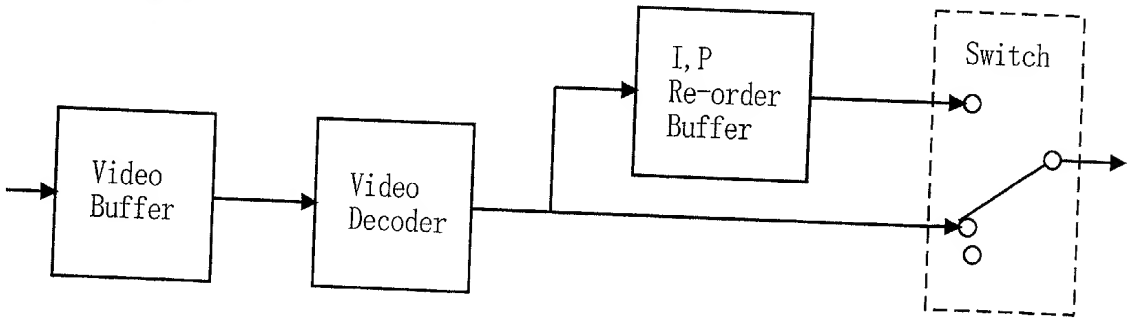


FIG. 19B
PRIOR ART

FIG. 19C
PRIOR ART



F I G. 2 0 PRIOR ART

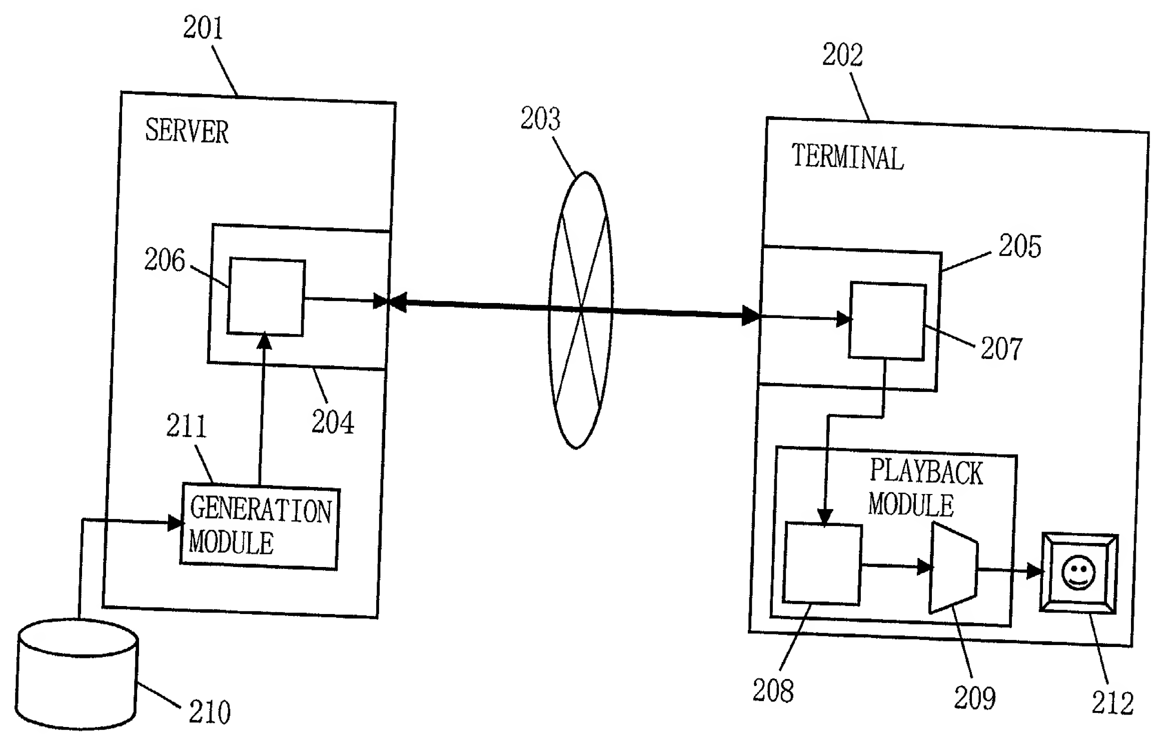


FIG. 21A
PRIOR ART

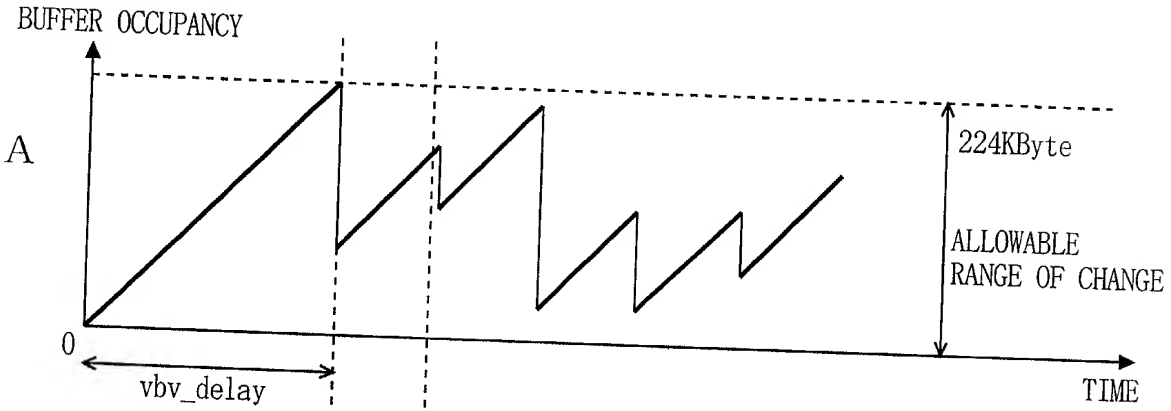


FIG. 21B
PRIOR ART